THE ASPIC HANDBOOK

A GUIDE TO

AUTHORS' SYMBOLIC PRE-PRESS INTERFACING CODES

British Printing Industries Federation

January 1984

Foreword

This booklet was prepared by the BPIF Development and Technology Committee's Composition Group, with a view to simplifying the relationship between an author or customer with word processor or computer, and the person doing the typesetting.

The relatively short list of codes recommended in the booklet is intended to complement the work being done on general mark-up languages by a number of other organisations. ASPIC is designed to satisfy an urgent need for an industry standard, and does so by providing a concise range of codes for the most popular types of formatting for typesetting. A comprehensive scheme, covering all the nuances of typesetting, will take much longer to develop, and must be related to the development of national and international standards for text processing. The Printing Industry Research Association is preparing the groundwork for a substantial standardisation effort, with implications for publishing via electronic and other new media as well as via typesetting and printing. Pira intends that any new schemes they develop will not conflict with ASPIC, and the BPIF is supporting Pira in this work.

The Group would like to record its thanks to Tony Randall of Electronic Village Limited, Richmond, for his considerable assistance in preparing the code and for allowing the term ASPIC to be used in this publication, and to John Birkenshaw who represented Pira.

Introducing ASPIC

ASPIC stands for Authors' Symbolic Prepress Interfacing Codes. It consists of a number of basic, non-technical codes or 'flags' which can be keyed with text on an input device, which indicate to the compositor or reading device any change in typeface, layout etc required by the author at a specific point, and which simplify subsequent setting. Such encoded text can be input on any word processor or micro computer and can be output through any photo-typesetter or non-impact printer with a programmable interface. ASPIC can be used before a decision has been made about the final typographic style, so an author can put the codes in before the publisher has settled on a style or decided how to print.

'Essential ASPIC' is a very short list of codes (see page 9) sufficient to cater for most jobs, which often only consist of text with paragraphs, headings and perhaps italics and bold as well as roman type. Most authors would get used to keying these codes with their copy in a very short time.

'Standard ASPIC' (see page 10) goes considerably further, catering for additional font changes including small caps, indented text, justified text, adjusting vertical and horizontal space and allowing space for illustrations. However, since the codes are logical they are easy to remember, and anyone regularly keying work should be fully conversant with them within days.

'Language ASPIC', 'Tabular ASPIC' and 'Mathematical ASPIC' are additional series of codes (see pages 11–12) which cater in a basic way for those specific requirements. They are not meant to be comprehensive and to avoid problems anyone keying such types of job should agree any further codes or systems necessary with their printer or typesetter before starting work. 'Supplementary ASPIC' is an additional short list of frequently used signs (see page 11).

Practically all the codes consist of two unshifted lower case letters surrounded by square brackets, the latter being known as opening and closing delimiters. These delimiters, since they appear on the visual display unit of the input device, allow one to make a clear distinction between a code, or instruction, and what is to be set; everything inside a set of brackets is a code, every-

thing outside is text. The codes will later be stripped out as they are converted into formatting commands for the output device.

Because square brackets were chosen as delimiters, it is necessary to have two codes to indicate square brackets where a job incorporates them (see page 12).

ASPIC uses the natural hierarchy of headings (such as chapter, section and subsection headings) and of text (such as main text, extracts and footnotes) to indicate what codes are inserted where.

Headings and text are specified by 'h' and 't' respectively, each one followed by a number and surrounded by the delimiting square brackets. Thus a chapter heading is specified by **[hl]** and a section heading by **[h2]**, while **[t1]** refers to main text and **[t2]** to extracts. To end some instructions 'x' is used, for example **[1x]** is used after a chapter heading and **[2x]** to end a section heading.

All such codes or flags in copy or recorded on tape or disc will be recognised by the compositor or reading device of the typesetter as a specific instruction. For instance **[h1]** may mean 'set the following heading in 14 point Garamond bold with 18 point line spacing over a measure of 18 picas', while **[1x]** could mean 'set the heading left justified, add a line space and then add an extra 6 points of leading'.

Authors should remember:

1. Use the codes or flags suggested in this publication wherever possible and tell your printer or typesetter that you are using ASPIC. Where you need to use supplementary non-standard codes draw up a list with your typesetter and identify them clearly as non-standard by asterisking.

2. Time spent in careful preparation and consultation with your typesetter, before you start keying, will save time and money at the interfacing stage.

3. Word processor formatting commands may conveniently be used to display text on screen and on hard copy printout; these will be removed automatically at interfacing.

4. The normal typing habit of putting two word spaces after a full stop at the end of a sentence, but within a paragraph, can still be followed. These will be converted automatically into a single word space, which is standard printing practice.

5. Word spaces should not be put after text paragraph ending codes, as these will be added to any indent at the beginning of the following paragraph.

6. In normal printing practice many signs, such as those for em and en dashes, plus, minus, multiplication and division, are not surrounded by word spaces. When keying therefore, the codes for such signs should not be preceded or followed by word spaces.

7. The global search and replace functions on word processors or microcomputers may be used to good effect; for example 'space, x, space' may be replaced by the code for multiplication, [*].

8. The lower case letter 'l' must not be used for the figure '1' and similarly the letter 'o' for zero.

9. It is convenient to use starting and ending codes eg. **[start]** and **[end]** at the beginning and end of each section of work. The 'start' command provides the parameters for any key line you may require before entering text (eg. 10/12 Times Roman, flushed left), sets the widest measure in the section that follows, and defines the indent, measure and justification of indented text. The 'end' command cancels these instructions and adds space before the next section is started.

10. Codes may be put in either when the text is being written, or after a draft has been prepared. The former has the attraction of avoiding one whole editing stage, while the latter permits the creative process to go on uninterrupted. The preferred method may depend on the type of work but, whichever is chosen, a carefully planned trial should be carried out when interfacing for the first time.

11. It is useful to provide your typesetter with a 'keyboard dump' showing all the characters on you word processor keyboard. This will allow you to use certain characters as direct input, for instance common fractions such as ½ and ¼, which will be recognised by the typesetter.

Detailed ASPIC

Wherever possible the meanings of codes included in this section are given both in language readily understandable to the word processor operator, and in computer typesetting terms.

Code	Word processing	Typesetting
Headings		
[h1]	Start of first level	Drop to baseline of beginning of heading and apply required justification (ie. centre, flush left or justify both sides); set measure, face, point size and line spacing within headings; determine hyphenation logic to be used and any reduction in inter-letter or -word spacing.
[1x]	End of first level heading, carriage return, line feed	Restore inter-letter and -word spacing to normal; enter quad left if heading start specified justified both sides; typesetting equivalent of carriage return and line feed; specify change in drop to next base line, other than ordinary line feed.
[h2]	Start of second level heading	As first level of heading [h1].
[2x]	End of second level heading	As first level of heading [1x].

As many levels of heading as are required may be specified.

Code	Word processing	Typesetting

Text

[t1]	First level text	Specifies indention of text or measure if different from before; sets face, point size, line spacing within text; text justification; determines if typesetter's hyphenation logic to be used.
[t2]	Second level text eg. extracts	As first level of text [t1].

There is no limit to the number of levels of text that can be specified in this way. These codes are normally used to specify changes in the text size, line spacing or face. They can also be used to specify different levels of indent as well, so cutting out the need for using the indent codes described in the following section. They are cancelled not by an 'x' command as with headings, but by others eg. [t2] or [h2] may be used to close [t1].

Indented text

[i1]	First level of indent	Ensures all subsequent text is set indented by a specified amount from the left-hand margin until cancelled by: (a) A text code, eg. [t1], when this cancels all indents or specifies a new indent. (b) Another indent code, eg. [i2]. (c) The 'end indent' code [ix].
[i2]	Second level indent	Indents subsequent text by another specified amount from the left-hand margin (not from the previous indent).
[ix]	Ends indent	Returns to left-hand margin (without line feed)

There is no limit to the number of levels of indent that can be specified in this way. As stated in the previous section, indents can be specified with text codes instead of a separate lot of indent codes.

Font changes

[r]	Roman	Change to roman face.
[i]	Italic	Change to italic face.
[b]	Bold	Change to bold face.
[bi]	Bold italic	Change to bold italic face.
[sc]	Small caps	For faces where a separately-cut small caps font does not exist, specifies the first level text face in a smaller type size, generally with a cap height similar to the x-height of the first level text size.
[sx]	Ends small caps	For faces where a separately-cut small caps face does not exist, returns to the point size of first level text.
[f#]	User-defined face, with # being any number.	Thus in a reference work containing both roman and sans serif faces, [r] might specify Plantin, [i] might specify Plantin italic, [b] might specify Plantin bold, [bi] might specify Plantin bold italic, [f1] might specify Helvetica and [f2] might specify Helvetica bold.

The code that starts a section of heading or text specifies the face in which the first character after the code is to be set. Only when you wish to change into another face do you have to use one of the codes above. Thus if you wish to change to italic, enter [i]. Then when you wish to change back to roman, enter [r].

continued on page 13

ASPIC

Authors' Symbolic Pre-press Interfacing Codes

Essential ASPIC

Headings

[h1] *Starts first level heading
[1x] Ends first level heading
[h2] Starts second level heading
[2x] Ends second level heading

Text

[t1] *Starts first level text
[t2] Starts second level text

Fonts

[r] Roman
[i] Italic
[b] Bold

Paragraph Endings

]] New paragraph, without indent
[ ] New paragraph, with indent

*There is no limit to the number of levels that may be specified in this way.

Standard ASPIC

Headings

[h1] *Starts first level heading
[1x] Ends first level heading
[h2] Starts second level heading
[2x] Ends second level heading

Text

[t1] *Starts first level text
[t2] Starts second level text

Indented text sections

[i1] *Starts first level indent
[i2] Starts second level indent
[ix] Ends indent

Font changes

[r] Roman
[i] Italic
[b] Bold
[bi] Bold italic
[sc] Small caps
[sx] Ends small caps
[f#] User-defined faces

Justification changes

[jc] Centred
[jj] Justified
[jl] Left (ragged right)
[jr] Right (ragged left)

Ending text paragraphs

]] New paragraph, without indent
[ ] New paragraph, with indent
[lc] Line centre, plus line feed
[lr] Line right, plus line feed
[[New line

Adjusting space

[+#] Adds # points, vertical
[-#] Subtracts # points, vertical
]] Blank line
[em] Em space, horizontal
[en] En space, horizontal

Dashes and quotes

[md] Em dash
[nd] En dash
' Closing single quote or apostrophe
'' Opening single quote

Superiors and inferiors

['#] Superior figure or letter #
[,#] Inferior figure or letter #

Gaps for illustrations

[qp] Space for quarter page
[hp] Space for half page
[wp] Space for whole page
[g#] Gap of fixed depth
[g?] Gap of unknown depth
[cs] Captions starts
[cx] Caption ends

Specialised codes

[id] Insert leader dots
[ir] Insert baseline rule
[is] Insert space
[iv] Insert vertical space
[p#] Pi character: refer to typesetter's list
[?#] Unknown pi character

= variable (usually numeric)

*There is no limit to the number of levels that may be specified in this way.

Supplementary ASPIC

LANGUAGE ASPIC

Accents

[ac] acute: for capital letter, enter [AC]
[an] Angstrom (Scandinavian a)
[ce] cedilla
[ci] circumflex
[gr] grave
[so] Slashed o (Scandinavian o)
[ti] tilde
[um] umlaut

Greek etcetera

[gA] Alpha A
[ga] alpha α
[gb] beta β
[gd] delta δ
[ge] epsilon ε
[gf] phi φ
[gg] gamma Υ
[gh] heta η
[gi] iota ι
[gk] kappa κ
[gl] lambda λ
[gm] mu μ
[gn] nu ν
[go] omicron ο
[gp] pi π
[gq] omega ω
[gj] rho ρ
[gs] sigma ς
[gt] tau τ
[gu] upsilon υ
[gv] theta θ
[gw] psi ψ
[gx] chi χ
[gy] xi ξ
[gz] zeta ζ
[sz] Esszett sign ß

SUPPLEMENTARY ASPIC

[ad] Arrow down ↓
[al] Arrow left ←
[ar] Arrow right →
[au] Arrow up ↑
[as] Asterisk *
[da] Dagger †
[dd] Double dagger ‡
[st] Star ★
[at] At sign @
[bs] Back slash \
[bu] Bullet ●
[co] Copyright sign
[ha] Hash #
[pa] Paragraph ¶
[se] Section mark §
[de] Degree sign °
[ft] Foot sign '
[in] Inch sign ''

continued overleaf

For Greek letters used in mathematical work, see 'Language ASPIC'
For superiors and inferiors see 'Standard ASPIC'

TABULAR ASPIC

Speak with your typesetter before using these

Defining columns

[kl] Key line starts
[cc#,α] Defines unspecified column
[kx] Key line ends

Calling and closing columns

[c] Calls next column
[c#] Calls column #
[cr] Carriage return

MATHEMATICAL ASPIC

[+] Plus sign +
[-] Minus sign −
[*] Multiplication sign ×
[/] Division sign ÷
[+-] Plus/minus ±

[bl] Brace, left {
[br] Brace, right }
[sl] Square bracket, left [
[sr] Square bracket, right]

[lt] Less than <
[mt] More than >
[le] Less than or equal to ⩽
[me] More than or equal to ⩾

[ig] Integral sign ∫
[gS] Sigma Σ
[sq] Square root √

For Greek letters used in mathematical work, see 'Language ASPIC'

For superiors and inferiors see 'Standard ASPIC'

Justification changes

[jc]	Centred	In practice there is seldom any need for these codes, as justification is often incorporated in heading and text commands.
[jj]	Justified both sides	
[jl]	Flush left (with ragged right)	
[jr]	Flush right (with ragged left)	

These codes affect the text following, not preceding, them. When you want, for example, a couple of lines of text centred, enter **[jc]** before the first character in those lines, and then **[jj]** – if you were in justified setting beforehand – before the first character of the text that follows.

Ending text paragraphs

]]	New paragraph without indent (usually placed at end of paragraph)	Flushes line left and starts new line
[]	New paragraph with indent (normal end of paragraph where next paragraph requires normal indent (usually placed at start of first line of new paragraph)	As above and indents next paragraph by a specified amount, usually one em.
[lc]	Line centre, with new line	Centres final line of paragraph, carriage return, line feed.
[lr]	Line right, with new line	Flushes final line of paragraph right, carriage return, line feed.
[[	New line (usually placed at end of paragraph)	Needed rarely, for instance when forcing new line in a heading.

These codes are not needed at the end of headings, as the appropriate manner of dealing with the white space remaining is specified in the end of heading code.

Adjusting space

[+#]	Adds # points of vertical space, where # is any whole number between 1 and 99	Drops the base line, on which the foot of capital letters rest, by # points.
[−#]	As above, but subtracts vertical space	
]]	Adds a blank line	
[em]	Adds one em of horizontal space; this is the size of the normal indent at the beginning of a paragraph	
[en]	Adds one en of horizontal space; this is the width of a number, and so it can be a useful code in tabular work	

The code **[em]** would not normally be used to indent new paragaphs as **[]** suffices.

Dashes and quotes

[md]	Em dash, like this: —
[nd]	En dash, like this: –
'	Apostrophe or closing single quote (key twice for closing double quotes)
"	Opening single quote (key twice for opening double quotes)

The last two codes are necessary because in print there is a distinction between opening and closing quotes, whereas on a word processor or micro, foot and inch signs are used indiscriminately as opening and closing single and double quotes. Thus, in order to achieve a printed "Good morning, Minister", enter on the word processor ""Good morning, Minister''

The codes for foot and inch signs are **[ft]** and **[in]**.

Accents

Code	Word processing	Typesetting
[ac]	Acute accent, thus: á. To access a higher accent to suit a capital letter, enter [AC]	Typesetters should make clear in their notes to customers on ASPIC whether accent codes are to be keyed before or after the letter to be accented eg. [ac]ecole or e[ac]cole.
[an]	Scandinavian å	
[ce]	Cedilla, thus ç	
[ci]	Circumflex accent, thus ê	
[gr]	Grave accent, thus: ì	
[so]	Scandinavian ø	
[ti]	Tilda	
[um]	Umlaut accent, thus: ö	

Accent codes have to be keyed either before or after the letter to be accented to suit the typesetting system. Keying before one knows the requirement of the typesetting system does not limit the customer to particular firms but where much of the work is accented there may be additional costs in converting from before-character accenting to after-, or vice versa.

Superiors and inferiors

['#]	Superior figure or letter	The single quote followed by a number or letter sets that character in the same face as the text but in a smaller size (eg. 6 point), a set distance (eg. 4 points) above the base line. It then reverts to the face and size of first level text; if working in any other level of text, it may well be necessary to re-specify it after using any superiors or inferiors.
[,#]	Inferior figure or letter	A comma followed by a number or letter sets that character in the same face as the text but in a smaller size (eg. 6 point), a set distance (eg. 2 points) below the base line.

Greek

[gA]	Greek alpha as a capital letter
[ga]	Greek alpha as a lower case letter

Most codes standing for Greek letters are phonetic but there are exceptions and in particular the use of **[gj]** for 'rho' should be noted, as **[gr]** is used to denote the grave accent.

Tabular work

Tabular work is difficult to set general standards or codes for, as the methods of handling it on different machines vary considerably. ASPIC therefore only sets basic codes for it; the use of additional codes may be warranted but in either case there must be close co-operation between typesetter and customer if time and money are not to be wasted. Codes will most often be used when dealing with repetitious tabular matter where the definitions of the columns are known beforehand.

Defining columns

[kl]	Key line starts. The key line is the line keyed on the word processor before starting a table where the indents and measures of each column are not known.	The computer on the typesetter uses the information in this line to calculate these automatically.
[cc#,∝]	Defines unspecified column. During the entering of the key line, begins and ends the widest amount of text or data required in a given column. The digit # indicates the column number: the letter ∝ indicates the justification required within the column.	The letter ∝ can be c, j, l or r depending on whether it is to be centred, justified, ranged left or right.
[kx]	Ends key line	

Calling and closing columns

[c]	Calls the next column or, after the carriage return code, calls the next column.	
[c#]	Calls a column by number. Typesetting systems vary in their capacity to handle columns and in the way they handle them, and typesetters must therefore be consulted before putting numbers to tables.	Typesetters should make it clear in notes to customers on ASPIC how tables of a repetitious nature and different formats are to be catered for.
[cr]	Carriage return; or absolute left-hand margin; no line feed.	

Gaps for illustrations

[qp]	Space for quarter page. This, and the following two codes, are used when a gap is needed in the text to allow a figure, table or illustration to be dropped in later.	The drop between the baselines on either side of the gap will be one quarter of the distance between the top and bottom baselines of the text on a normal page.
[hp]	Space for half page	
[wp]	Space for whole page	
[g#]	Gap of fixed depth. The hash mark may represent any number up to 999, and this code will open up a gap of this number of points.	
(g?]	Gap of unknown depth. This code is entered to enable the entire text to be searched, whether on the word processor or on the phototypesetter, once the size of gaps are known and the appropriate size of gap manually entered in each case.	

Gaps for illustrations *continued*

[cs] Caption starts. This is a specialised code that is needed only when it is necessary to link the text unbreakably to what precedes it in the pagination process.

[cx] Caption ends. This needs to be followed by the code for the appropriate level of text.

Specialised codes

[id] Insert leader dots. This code fills up the space, either to the right-hand margin or to the next piece of text on the line, with dots like these .....

[ir] Insert baseline rule. This code fills up the space, either to the right-hand margin or to the next piece of text on the line, with a series of rules like this _____ The author, when in doubt about the thickness of the rule, should consult the typesetter; the thickness is often one twelfth of the point size of the text.

[is] Insert space. This code is used when the space remaining at the end of a line needs to be put, as in a contents list, at some point in the middle of the line.

[iv] Insert vertical space. This code is used when it is desirable to indicate that at certain places, eg. above and below figures, extra space up to a specified limit may be added during the pagination process.

continued overleaf

DATE DUE

Specialised codes *continued*

[p#]	Pi character. A pi character is any character that is not on the typesetter keyboard or on the list of ASPIC supplementary characters.	The number entered is the number on the typesetter's current list of pi characters.
[?#]	Pi character, when no reference to it is made in the typesetter's list. For each such character, a different number will follow the question mark, and in the typographical specification accompanying the job, these characters will be defined.	

British Printing Industries Federation
11 Bedford Row, London WC1R 4DX
Telephone 01-242 6904